2N-02

マイクロブログ情報解析によるイベント検出手法の提案

吉澤康太[†1]　秋岡明香[†2]
明治大学大学院[†1]　明治大学[†2]

1 はじめに

近年はICT(情報通信技術)の普及・発展に伴い、様々な電子情報端末（スマートフォンやタブレット等）が日常生活を送る上で必要不可欠になってきている．また，情報端末からのSNS利用が一般化したことでSNSユーザーの話題や流行を分析・予測することは社会分析やマーケティングにおいて重要となってきている．本稿では，膨大な量の情報を持つマイクロブログである Twitter[1] の投稿情報のみに着目し，教師なし機械学習によりリアルタイムに行われているイベント情報を抽出する手法を提案する．従来の手法では、単語の出現頻度やイベント情報の投稿に着目して機械学習をすることでイベントの特定をしていた．しかし，本稿では fastText[2]を用いることで単語の意味的なつながりを加味することでより正確なイベントの特定を試みる．提案手法の有効性を確認するため，抽出された単語がどの程度イベント単語と認識できるかについて評価を行う．

2 関連研究

中澤らは Twitter の位置情報付きツイートに着目し，クラスタリングを行うことでツイートの密集地点を割り出し，TF-IDFによってイベントを抽出する手法を提案している[3]．しかし，位置情報付きツイートはその数が非常に少なく，中澤らの研究では一つのランドマークにつき１０ツイート程しか抽出できないことが判明している．本稿では位置情報付きツイートのみでなく，対象の地域名を含むツイート全てに対して解析を行う．また，TF-IDF のみではイベント単語の表記揺れに対応ができないことが判明している．本稿では fastText を使用することでイベント単語の表記揺れに対応していく．

他にも，イベントの告知に関するツイートに着目することでイベント情報を特定する手法が提案されている[4]．この手法では，あらかじめ用意した地名のリストとツイートを関連づけ，Support Vector Machine によってイベント情報を抽出している．本稿ではイベントに関する投稿だけでなく，指定地域の単語を含むツイート全てに対して解析を行うものであるため，告知の行われていないイベント情報にも対応ができると考える．

3 提案手法

本稿では収集したツイートに対して教師なし学習を行うことでリアルタイムに起きているイベントを抽出する．提案手法のフローは以下の通りである．

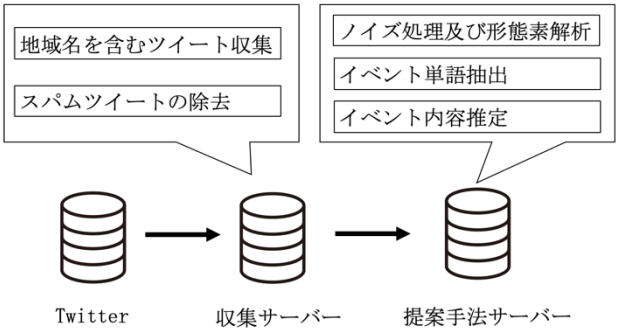

図１．提案手法の処理フロー

(1) 特定の地域名を含むツイートを収集し，ノイズ処理及び形態素解析を行う．
(2) 収集されたツイートからイベント単語を抽出する．
(3) 抽出された単語からイベント内容を推定する．

Twitter では不特定多数のユーザーが様々なツイートを投稿している為，イベントに関するツイートをピンポイントで収集することは不可能である．そこで,イベントの発生が起こりやすいと考えられる各都道府県の人口上位５市町村名をキーワードとしてツイートを収集する．上記(1)で特定の地域名を含むツイートを収集するにあたり,これらの地域名をクエリとして Twitter Search API でツイートの収集を行う．その際，リツイート，URL，スパムアカウントによるツイートはノイズとして除去する．次に，収集したツイートに対して絵文字や数字，ストップワードの除去を行う．その後，前処理が完了したコーパスに対し形態素解析を行う．形態素解析には MeCab [5]を利用し，辞書には俗語に強い Neologd [6]を使用した．

形態素解析が完了したコーパスに対して fastText で機械学習をさせ，イベント単語の抽出を行う．fastText では Skip-gram を使用し，ある単語から周辺の単語の共起確率を計算してベクトル化している．イベントに関するツイートではイベント単語とイベント発生場所を表す単語が共起している可能性が高い．また，fastText が持つ subword は，単語の活用形を考慮することで他の類似している単語も意味的に近いとみなす．こうした特性を利用することで，ツイート中で省略されている，いわゆる表記揺れのあるイベント名称を推定していく．

*Proposal of event extraction method by microblog information analysis
†1 KOTA YOSHIZAWA, Meiji Graduate school
†2 SAYAKA AKIOKA, Meiji University

(3)では検索に用いた地域の付近にあるランドマーク名と類似度の高い単語を抽出する．具体的には，検索に用いた地域の付近のランドマークを Yahoo! Open Local Platform [7] の場所情報 API によって取得し，ランドマークと類似度の高い単語上位５つをキーワード群として地域名とラベル付けをする．

4 評価実験

提案手法によりイベント単語をどの程度抽出可能かを調査するために実験を行った．対象とする地域名は表１の通りである．抽出した単語がイベント単語かどうかの判定は１地域あたり 3 人の被験者が評価を行った．それぞれの地域と抽出されたイベント単語について以下の 3 段階で評価値を付与した．

（２）：キーワード群からイベントと認識できる．
（１）：イベントと認識できない．
（０）：関係のない単語が抽出されている．

5. 考察

イベント単語の抽出結果を表２に示す．東松山と前橋に関しては高評価を得たオベント単語が多く，適切なイベント抽出結果が得られた．特に東松山では先日の台風 19 号の影響で営業を停止していた大型ショッピングモールであるピオニーウォークの営業再開というイベントを，また，前橋においては前橋初市まつりというイベントを検出することができた．また，それぞれ抽出されたイベント関連単語のすべてが fastText の学習結果において類似度 0.9 を超えていた．適切に抽出できた理由としては，両地域に共通して言える特徴にローカル性があり，ノイズツイートが少ないことが挙げられる．しかし，川崎と柏に関しては期待した結果が得られなかった．イベントが正しく検出できなかった原因として，川崎は地域名だけでなく人名として使われている場合でも検索対象となってしまい，その結果人名としての川崎が収集されたツイートに多く含まれてしまったことがあげられる．また，柏は日本のサッカーチームである柏レイソルとしての特徴が色濃くなっており，サッカーに関連する単語が多く抽出されている．これは定期的にJリーグに関する情報を発信しているアカウントがノイズとなっており，ツイートの前処理が不十分であったことが考えられる．

6. おわりに

本稿では特定地域名を含むツイートを教師なし学習をすることによりイベント単語を抽出する手法を提案した．評価実験により，ノイズの少ないローカル地域についてこの提案手法は有効であることを確認した．今後の課題として，ノイズとみなすツイートの条件の追加，およびツイート数が多い地域への対策としてイベント単語と共起しやすい単語リストによるフィルターの追加などの改善策が挙げられる．

表１．実験対象地域名の一覧

都県名	地域名
東京	世田谷区，練馬区，大田区，足立区，江戸川区
千葉	千葉，船橋，松戸，市川，柏
埼玉	川越，東松山，浦和，大宮，所沢
神奈川	横浜，藤沢，相模原，川崎，横須賀
茨城	つくば，水戸，日立，ひたちなか，土浦
栃木	宇都宮，小山，栃木，足利，佐野
群馬	高崎，太田，前橋，伊勢崎，桐生

表２．イベント単語抽出結果

高評価(総評価 6)		低評価(総評価１以下)	
東松山	前橋	川崎	柏
台風	初市	川崎選手	ジェフ
ピオニ	だるま市	川崎離脱	レイソル
再開	群馬大学	川崎和馬	レッズ
被災	前橋プラザ	川崎芽衣子	アビスパ
休止	群馬	ラゾーナ	教え子

謝辞 本研究は JSPS 科研費 16KK0008 の助成を受けたものである．

参考文献

[1] Twitter : https://twitter.com
[2] Piotr Bojanowski, Edouard Grave, Armand Joulin, Tomas Mikolov, “Enriching Word Vectors with Subword Information” (2016)
[3] 中澤昌美，池田和史，服部元，小野智弘，“位置情報付きツイートからのイベント検出手法の提案”(2012) 情報処理学会第７４回大会
[4] 山田渉，菊地悠，落合桂一，鳥居大祐，稲村浩，太田賢，“マイクロブログを用いたイベント情報抽出技術” 情報処理学会論文誌 vol57 No.1 pp123-132(Jan. 2016)
[5] Taku Kudo, Kaoru Yamamoto, Yuji Matsumoto: Applying Conditional Random Fields to Japanese Morphological Analysis, Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004), pp.230-237 (2004.)
[6] mecab-ipadic-neologd : https://github.com/neologd/mecab-ipadic-neologd/blob/master/README.ja.md
[7] YOLP:https://developer.yahoo.co.jp/webapi/map/openlocalplatform/v1/placeinfo.html